



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE

United States Patent and Trademark Office

Address: COMMISSIONER FOR PATENTS

P.O. Box 1450

Alexandria, Virginia 22313-1450

www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/716,622	11/20/2003	Akira Watanabe	Y2238.0054	6336
32172	7590	06/17/2010		
DICKSTEIN SHAPIRO LLP				
1633 Broadway				
NEW YORK, NY 10019				
EXAMINER				
HOTELLING, HAROLD A				
ART UNIT		PAPER NUMBER		
2164				
MAIL DATE		DELIVERY MODE		
06/17/2010		PAPER		

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.



UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

**BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES**

Application Number: 10/716,622
Filing Date: November 20, 2003
Appellant(s): WATANABE, AKIRA

Joseph W. Ragusa (Reg. No. 38,586)
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed March 19, 2010 appealing from the Office action mailed October 8, 2009.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The examiner has no comment on the appellant's statement of the status of amendments after final rejection contained in the brief.

(5) Summary of Claimed Subject Matter

The examiner has no comment on the summary of claimed subject matter contained in the brief.

(6) Grounds of Rejection to be Reviewed on Appeal

The examiner has no comment on the appellant's statement of the grounds of rejection to be reviewed on appeal. Every ground of rejection set forth in the Office

action from which the appeal is taken (as modified by any advisory actions) is being maintained by the examiner except for the grounds of rejection (if any) listed under the subheading "WITHDRAWN REJECTIONS." New grounds of rejection (if any) are provided under the subheading "NEW GROUNDS OF REJECTION."

(7) Claims Appendix

The examiner has no comment on the copy of the appealed claims contained in the Appendix to the appellant's brief.

(8) Evidence Relied Upon

- Li (U.S. patent number 6,754,662) (assigned to Nortel) (effective filing date: August 1, 2000)
- Kottisa (US 2004/0049494 A1) (assigned to Hewlett-Packard) (effective filing date: September 10, 2002)

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

35 U.S.C. §103 rejection

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the

invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

The factual inquiries set forth in *Graham v. John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

Claims 1, 4 – 9, and 12 – 17 (assigned to NEC) (effective filing date: November 20, 2002) are rejected under 35 USC 103 (a) as being obvious over Li (U.S. patent number 6,754,662) (assigned to Nortel) (effective filing date: August 1, 2000) in view of Kottisa (US 2004/0049494 A1) (assigned to Hewlett-Packard) (effective filing date: September 10, 2002).

With respect to independent claim 1, Li teaches **[a] packet search device that performs packet filter search for an inputted packet** (column 2, lines 13 – 14: “The present invention relates to a method and apparatus for classifying data packets.”), **comprising:**

a first search processor that searches predetermined conditional statements corresponding to a plurality of information areas included in header information of said packet using a first search method to generate first search results (column 4, lines 23 – 30: “Forwarding engine 102 first attempts to retrieve a classID for a defined flow corresponding to the incoming packet from cache 108 by

calculating a hash key (block 304) and using it to lookup a corresponding entry in cache 108 (block 306). If the classID cannot be found in cache 108 (determined in block 308), forwarding engine 102 performs a search on stored classification information in memory 110 (block 310).”; and a **second search processor** (column 3, lines 35 – 41: “As shown in FIG. 1, there are two main engines in the example architecture: a forwarding engine 102, which performs the search for an appropriate classID for arriving packets, and a control engine 104, which is used to calculate the cache hit statistics and to maintain the cached hash table. In one example of the invention, each engine is implemented by a separate CPU, . . .”) . . .

wherein said first search processor divides said packet header information into a plurality of information areas and searches across each search conditional statements structured as binary search trees for each of said information areas separately (column 5, lines 54 – 57: “the choice of data structures (i.e. a link list or a binary tree or other structure) may depend on the particular design objective of the packet classifier.”), and

wherein said second search processor searches aggregated search results of said first search processor using a Hash method (column 3, lines 38 – 39: “a control engine 104 . . . is used to calculate the cache hit statistics . . .”) (column 4, lines 7 – 9: “Cache 108 is preferably comprised of high-speed cache memory that is readily available to forwarding engine 102 and control engine 104. Cache 108 stores a hash table . . .”).

Li does not appear to explicitly teach (but Kottisa does teach) **that searches the**

first search results of said first search processor using a second search method that is different from said first search method (paragraph [0006]: "According to an aspect of the present invention, there is provided a method for traversing search results from a search query. The method includes receiving the search results arranged in a first order and allowing traversing of the search results in a selected second order different from the first order."), . . .

Li and Kottisa are analogous art because they are from the problem-solving area of searches. At the time of the invention, it would have been obvious to one of ordinary skill in the art, having the teachings of Li and Kottisa before him or her, to modify the "control engine 104" of Li to include the "traversing of the search results in a selected second order different from the first order" of Kottisa because the combination makes more search results reachable.

The suggestion for doing so would have been Kottisa (paragraph [0022]: "Advantageously, the method for traversing search results described above gives a user a chance to reach search results that were otherwise unreachable or reachable only after tedious traversing using prior art means.").

Therefore, it would have been obvious to combine Kottisa with Li to obtain the invention as specified in claim 1.

With respect to dependent claim 4, Li teaches **[t]he packet search device according to claim 1, comprising a search database for managing each search result of said first and second search processors for each of said information**

area (column 4, lines 9 – 13: “Cache 108 stores a hash table with entries filled by class of service identifiers (i.e. classIDs) for (generally) the most recently detected flows. These entries are accessed by a hash key index that is generated by a hash function from packet header information . . .”).

With respect to dependent claim 5, Li teaches **[t]he packet search device according to claim 4, wherein said search database has a plurality of search keys** (column 4, lines 9 – 13: “Cache 108 stores a hash table with entries filled by class of service identifiers (i.e. classIDs) for (generally) the most recently detected flows. These entries are accessed by a hash key index that is generated by a hash function from packet header information . . .”).

With respect to dependent claim 6, Kottisa teaches **[t]he packet search device according to claim 1, wherein said second search processor manages only combinations of search results** (paragraph [0017]: “FIG. 3 illustrates a sequence 20 of steps for traversing the search results 18. The sequence 20 starts in a RESULTS AVAILABLE step 22, wherein the Web search engine 2 receives the search results 18 in a first order. This first order may be the result of ranking of the search results 18. The sequence 20 next proceeds to a PRESENT RESULTS step 24, wherein the browser 12 receives a first portion of the search results 18 from the Web search engine 2 . . .”).

With respect to dependent claim 7, Li teaches **[t]he packet search device according to claim 1, wherein at least QoS (Quality of Service) information and filter information are searched for based on said header information** (column 3, lines 55 – 60: “Memory 110 includes stored information about how different classes of network traffic are identified and how they are to be treated. Such information can include SLAs for DiffServ networks, and other filters and parameters for establishing different levels of Quality or Class of Service for different flows of traffic.”).

With respect to dependent claim 8, Li teaches **[t]he packet search device according to claim 1, wherein said packet search processing is performed at least in a router and a firewall** (column 3, lines 29 – 33: “FIG. 1 is a block diagram showing a classification architecture 100 in accordance with one example of the invention. Such an architecture can be provided in . . . an enterprise access/firewall router, a general Internet access router, etc.”).

With respect to independent claim 9, Li discloses **[a] packet processing search method, on a packet search device that includes a first search processor and a second search processor, that searches for a packet filter for an inputted packet before performing packet processing** (column 2, lines 13 – 14: “The present invention relates to a method and apparatus for classifying data packets.”) (column 3, lines 35 – 41: “As shown in FIG. 1, there are two main engines in the example architecture: a forwarding engine 102, which performs the search for an appropriate

classID for arriving packets, and a control engine 104, which is used to calculate the cache hit statistics and to maintain the cached hash table. In one example of the invention, each engine is implemented by a separate CPU, . . ."), **comprising:**

a first step of the first search processor searching predetermined conditional statements corresponding to a plurality of information areas included in header information of said packet using a first search method to generate first search results (column 4, lines 23 – 30: "Forwarding engine 102 first attempts to retrieve a classID for a defined flow corresponding to the incoming packet from cache 108 by calculating a hash key (block 304) and using it to lookup a corresponding entry in cache 108 (block 306). If the classID cannot be found in cache 108 (determined in block 308), forwarding engine 102 performs a search on stored classification information in memory 110 (block 310)."); **and a second step of the second search processor** (column 3, lines 35 – 41: "As shown in FIG. 1, there are two main engines in the example architecture: a forwarding engine 102, which performs the search for an appropriate classID for arriving packets, and a control engine 104, which is used to calculate the cache hit statistics and to maintain the cached hash table. In one example of the invention, each engine is implemented by a separate CPU, . . .") . . .

wherein in said first step the first search processor divides said packet header information into a plurality of information areas and searches across each search conditional statements structured as binary search trees for each of said information areas separately (column 5, lines 54 – 57: "the choice of data structures (i.e. a link list or a binary tree or other structure) may depend on the particular design

objective of the packet classifier."), and

wherein in said second step the second search processor searches aggregated search results of said first step using a Hash method (column 3, lines 38 – 39: "a control engine 104 . . . is used to calculate the cache hit statistics . . .") (column 4, lines 7 – 9: "Cache 108 is preferably comprised of high-speed cache memory that is readily available to forwarding engine 102 and control engine 104. Cache 108 stores a hash table . . .").

Li does not appear to explicitly teach (but Kottisa does teach) **searching the first search results at said first step using a second search method that is different from said first search method** (paragraph [0006]: "According to an aspect of the present invention, there is provided a method for traversing search results from a search query. The method includes receiving the search results arranged in a first order and allowing traversing of the search results in a selected second order different from the first order."), . . .

Li and Kottisa are analogous art because they are from the problem-solving area of searches. At the time of the invention, it would have been obvious to one of ordinary skill in the art, having the teachings of Li and Kottisa before him or her, to modify the "control engine 104" of Li to include the "traversing of the search results in a selected second order different from the first order" of Kottisa because the combination makes more search results reachable.

The suggestion for doing so would have been Kottisa (paragraph [0022]: "Advantageously, the method for traversing search results described above gives a user

a chance to reach search results that were otherwise unreachable or reachable only after tedious traversing using prior art means.”).

Therefore, it would have been obvious to combine Kottisa with Li to obtain the invention as specified in claim 9.

With respect to dependent claim 12, Li teaches **[t]he packet processing search method according to claim 9, wherein each search result at said first and second steps is managed for each of said information areas using a search database** (column 4, lines 9 – 13: “Cache 108 stores a hash table with entries filled by class of service identifiers (i.e. classIDs) for (generally) the most recently detected flows. These entries are accessed by a hash key index that is generated by a hash function from packet header information . . .”).

With respect to dependent claim 13, Li teaches **[t]he packet processing search method according to claim 12, wherein said search database has a plurality of search keys** (column 4, lines 9 – 13: “Cache 108 stores a hash table with entries filled by class of service identifiers (i.e. classIDs) for (generally) the most recently detected flows. These entries are accessed by a hash key index that is generated by a hash function from packet header information . . .”).

With respect to dependent claim 14, Kottisa teaches **[t]he packet processing search method according to claim 9, wherein in said second step the second**

search processor manages only combinations of search results (paragraph [0017]:

"FIG. 3 illustrates a sequence 20 of steps for traversing the search results 18. The sequence 20 starts in a RESULTS AVAILABLE step 22, wherein the Web search engine 2 receives the search results 18 in a first order. This first order may be the result of ranking of the search results 18. The sequence 20 next proceeds to a PRESENT RESULTS step 24, wherein the browser 12 receives a first portion of the search results 18 from the Web search engine 2 . . .").

With respect to dependent claim 15, Li teaches **[t]he packet processing search method according to claim 9, wherein at least QoS (Quality of Service) information and filter information are searched for based on header information in said packet** (column 3, lines 55 – 60: "Memory 110 includes stored information about how different classes of network traffic are identified and how they are to be treated. Such information can include SLAs for DiffServ networks, and other filters and parameters for establishing different levels of Quality or Class of Service for different flows of traffic.").

With respect to dependent claim 16, Li teaches **[t]he packet processing search method according to claim 9, said packet search processing is performed at least in a router and a firewall** (column 3, lines 29 – 33: "FIG. 1 is a block diagram showing a classification architecture 100 in accordance with one example of the invention. Such an architecture can be provided in . . . an enterprise access/firewall router, a general

Internet access router, etc.”).

With respect to independent claim 17, Li discloses **[a] computer-readable medium storing a program for causing a program-controlled packet search device that includes a first search processor and a second search processor to perform a packet processing search method that searches for a packet filter for an inputted packet before performing packet processing, causing the program-controlled packet search device to execute** (column 2, lines 13 – 14: “The present invention relates to a method and apparatus for classifying data packets.”) (column 3, lines 35 – 41: “As shown in FIG. 1, there are two main engines in the example architecture: a forwarding engine 102, which performs the search for an appropriate classID for arriving packets, and a control engine 104, which is used to calculate the cache hit statistics and to maintain the cached hash table. In one example of the invention, each engine is implemented by a separate CPU, . . .”),

first processing by the first search processor that searches predetermined conditional statements corresponding to a plurality of information areas included in header information of said packet using a first search method to generate first search results (column 4, lines 23 – 30: “Forwarding engine 102 first attempts to retrieve a classID for a defined flow corresponding to the incoming packet from cache 108 by calculating a hash key (block 304) and using it to lookup a corresponding entry in cache 108 (block 306). If the classID cannot be found in cache 108 (determined in block 308), forwarding engine 102 performs a search on stored classification

information in memory 110 (block 310)."); and **second processing by the second search processor** (column 3, lines 35 – 41: "As shown in FIG. 1, there are two main engines in the example architecture: a forwarding engine 102, which performs the search for an appropriate classID for arriving packets, and a control engine 104, which is used to calculate the cache hit statistics and to maintain the cached hash table. In one example of the invention, each engine is implemented by a separate CPU, . . .") . . .

wherein in said first processing the first search processor divides said packet header information into a plurality of information areas and searches across each search conditional statements structured as binary search trees for each of said information areas separately (column 5, lines 54 – 57: "the choice of data structures (i.e. a link list or a binary tree or other structure) may depend on the particular design objective of the packet classifier."), and

wherein in said second processing the second search processor searches aggregated search results of said first step using a Hash method (column 3, lines 38 – 39: "a control engine 104 . . . is used to calculate the cache hit statistics . . .") (column 4, lines 7 – 9: "Cache 108 is preferably comprised of high-speed cache memory that is readily available to forwarding engine 102 and control engine 104. Cache 108 stores a hash table . . .").

Li does not appear to explicitly teach (but Kottisa does teach) **that searches the first search results of said first processing using a second search method that is different from said first search method** (paragraph [0006]: "According to an aspect of the present invention, there is provided a method for traversing search results from a

search query. The method includes receiving the search results arranged in a first order and allowing traversing of the search results in a selected second order different from the first order.”), . . .

Li and Kottisa are analogous art because they are from the problem-solving area of searches. At the time of the invention, it would have been obvious to one of ordinary skill in the art, having the teachings of Li and Kottisa before him or her, to modify the “control engine 104” of Li to include the “traversing of the search results in a selected second order different from the first order” of Kottisa because the combination makes more search results reachable.

The suggestion for doing so would have been Kottisa (paragraph [0022]: “Advantageously, the method for traversing search results described above gives a user a chance to reach search results that were otherwise unreachable or reachable only after tedious traversing using prior art means.”).

Therefore, it would have been obvious to combine Kottisa with Li to obtain the invention as specified in claim 17.

(10) Response to Argument

The appellant’s arguments in the March 19, 2010 Appeal Brief have been fully considered, but they are not persuasive.

Regarding the cancelled dependent claim 2 limitation pulled up into independent claim 1, the appellant argues on pages 7 – 8:

Li at col. 5, lines 54-57 . . . may be said to show that the choice of whether to use a link list, or binary tree, or other structure, depends on the design objective, it says nothing whatsoever relating to what is explicitly recited in claim 1, i.e., that the first search processor divides the packet header information into a plurality of information areas and searches across each search conditional statements structured as binary search trees for each of the information areas separately.

The examiner respectfully disagrees with the appellant's belief that "Li at col. 5, lines 54 – 57 . . . says nothing whatsoever relating to what is explicitly recited in claim 1" because the Li "packet classifier" teaches the appellant's claim 1 **first search processor** and the Li "binary tree" teaches the appellant's claim 1 **binary search tree** (see Li, column 5, lines 54 – 57).

Regarding the cancelled dependent claim 3 limitation pulled up into independent claim 1, the appellant argues on page 9 regarding the Li reference:

Col. 3, lines 38 – 39 discusses the use of a hash table generally, but contains no teaching of using a Hash method to search aggregated search results from a previous search. Col. 4, lines 7 – 9 simply discusses that computer cache may contain a hash table. Thus, contrary to the position taken at page 5 of the Final Office Action, the Examiner has failed to identify any teaching of the feature of searching *the aggregated search results the first search processor* at all, still less using a Hash method to do so, as claimed.

The examiner agrees with the appellant's belief that Li, "Col. 3, lines 38 – 39 . . . contains no teaching of using a Hash method," but the combination of column 3, lines 38 – 39 and column 4, lines 7 – 9 from Li does teach the appellant's claim 1 **wherein said second search processor searches aggregated search results of said first search processor using a Hash method** (column 3, lines 38 – 39: "a control engine 104 . . . is used to calculate the cache hit statistics . . .") (column 4, lines 7 – 9: "Cache 108 is preferably comprised of high-speed cache memory that is readily available to

forwarding engine 102 and control engine 104. Cache 108 stores a hash table . . .”).

The appellant argues on page 9:

According to Kottisa, typical search results may include thousands of results that are not of any interest to the user. Kottisa's system arranges the search results of the search engine, originally presented in a first order, into a second order, with the hope that the new order will make it more likely that relevant search results will be seen by the user as he looks through the results.

In Kottisa, the *only* search performed is the one performed by search engine 2. No second search is performed at all. The very same results are simply *rearranged*, without searching them again. Kottisa's scrambling of the search results, in the hope that a user will spot more relevant results, does not amount to performing a new search on the results of the search engine. For this additional reason, even if Li and Kottisa are combined, they do not teach or suggest all of the elements recited in claim 1 or the other independent claims.

The examiner respectfully disagrees with the appellant's belief that “[n]o second search is performed at all” because, as the appellant points out, “Kottisa's system arranges the search results of the search engine, originally presented in a first order, into a second order, . . .”

Because the Kottisa system rearranges search results from a first order to a second order, the Kottisa system must perform a second search that searches the results of the first search in order to determine the search result that will take first place in the rearranged second order, the search result that will take second place in the rearranged second order, the search result that will take third place in the rearranged second order, etc.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/H. A. H./
Examiner, Art Unit 2164

Harold A. Hotelling
Examiner, Art Unit 2164

Conferees:

/Harold A Hotelling/
Harold A. Hotelling
Examiner, Art Unit 2164

/Charles Rones/
Supervisory Patent Examiner, Art Unit 2164

Charles Rones
Supervisory Patent Examiner, Art Unit 2164

/Hosain T Alam/
Supervisory Patent Examiner, Art Unit 2166

Hosain Alam
Supervisory Patent Examiner, Art Unit 2166